Express Mail mailing Label No. EL 898554309US
Date of Deposit Mon, Jul 9, 2001.

I hereby certify that this paper is being deposited with the United States Postal Service "Express Mail Post Office to Addressee" service under 37 CFR 1.10 on the date indicated above and is addressed to the BOX PATENT APPLICATION, Assistant Commissioner for Patents, Washington, D.C. 20231.

Abby Berghella
(Typed or printed name of person mailing paper or fee)

Abby Berghella
(Signature of person mailing paper or fee)

SOFT-OUTPUT DECODING METHOD AND APPARATUS FOR CONTROLLED INTERSYMBOL INTERFERENCE CHANNELS

The present invention claims priority under 35 U.S.C., Section 119 (e) from U.S. provisional application serial no. 60/239,254, entitled “Soft-Output Decoding Method and Apparatus for Controlled Intersymbol Interference Channels”, filed 12-October-2000, the contents of which are incorporated herein by reference.

BACKGROUND OF THE INVENTION

Field of the Invention

[0001] The present invention relates to a soft-output decoder for controlled intersymbol interference channels, and more particularly to an apparatus and a method for reducing complexity in a soft output Viterbi decoding algorithm used in partial response channels, such as those used to model magnetic recording channels.

Description of the Related Art

[0002] Transmission of digital data is inherently prone to interference that may introduce errors into the transmitted data. Error detection schemes have been suggested to determine as reliably as possible whether errors have been introduced into the transmitted data. One such scheme is known as convolutional coding.

[0003] Convolutional codes have been introduced to allow receivers of digital data to correctly determine the transmitted data even when errors may have occurred during transmission. The convolutional codes introduce redundancy into the transmitted data and pack the transmitted data into packets in which the value of each bit is dependent upon earlier bits in the sequence. Thus, when a few errors occur, the receiver can still deduce the original data by tracing back possible sequences in the received data.

[0004] Optimal decoding is often a very complex task, and may require large periods of time, which are not usually available. In order to overcome this problem, iterative decoding techniques have been developed. Rather than determining immediately whether a received bit is a zero or a one, the receiver assigns each bit a value on a multi-level scale representative of the probability that the bit is a one. A common scale is the logarithmic likelihood ratio (LLR). The LLR represents each bit by some integer in an interval such as, for example, {-32,31}. In this example, the value of 31 signifies that the transmitted bit was a one with very high probability, and the value of –32 signifies that the transmitted bit was a zero with very high probability, whereas the value of 0 indicates indeterminacy as to whether the transmitted bit was a zero or a one.

[0005] Data represented on the multi-level scale is referred to as “soft data”. Iterative decoding is usually soft-input/soft-output; i.e., the decoding process receives a sequence of inputs corresponding to probabilities for the bit values and provides as output corrected probabilities which take into account the constraints of the code. Generally, a decoder which performs iterative decoding uses soft data from former iterations to decode the soft data read by the receiver. The iterative decoding is carried out for a plurality of iterations until it is believed that the soft data closely represents the transmitted data.

[0006] Decoding techniques for convolutional codes are very well known in the literature and are the subject of many patents. For example, see U.S. Patent Nos. 6,182,261; 6,161,209; and 6,145,114, the contents of each of which are incorporated herein by reference. In addition, several commonly assigned copending patent applications relate to this subject matter. These include U.S. Patent Application No. 09/730,597, filed on December 7, 2000, and entitled “Address Generator for LDPC Encoder and Detector and Method Thereof”; U.S. Patent Application No. 09/730,752, filed on December 7, 2000, and entitled “LDPC Encoder and Method Thereof”; U.S. Patent Application No. 09/730,603, filed on December 7, 2000, and entitled “LDPC Detector and Method Thereof”; and U.S. Patent Application No. 09/730,598, filed on December 7, 2000, and entitled “Parity Check Matrix and Method of Forming Thereof”, the contents of all of which are incorporated herein by reference.

[0007] Reference is now made to Fig. 6, which is a block diagram of a data transmission system as described in more detail in commonly assigned copending application "Address Generator for LDPC Encoder and Decoder and Method Thereof", filed on December 7, 2000 and assigned application serial No. 09/730,957, the contents of which are incorporated herein by reference. In general as shown therein, a digital data transmission system comprises a transmitting section 300' for transmitting user data to receiver 500' via communication channel 401.

[0008] The operation of transmission section 300' will now be explained. Prior to processing by transmitting section 300', input or user data maybe encoded with an error correcting code, such as the Reed/Solomon code, or run length limited encoded (RLL) or a combination thereof by encoder 302. Addresses for the parity equations of linear block code encoder 304 are generated by address generator 328 in accordance with an index of the bits of data, the index being determined by address generator 328. Address generator 328 is responsive to counter 730 under the control of controller 740. Controller 740 synchronizes counter 730 to the output of encoder 302 so that counter 730 can provide a count of the number of bits in a codeword output by encoder 302 and a count of the number of codewords.

[0009] Linear block code encoder 304 utilizes the user data and address from address generator 328 to provide the parity bits to multiplexer 306. Linear block code encoder 304 is preferably implemented as a low-density parity-check code (LDPC) encoder as described in commonly assigned, copending patent application entitled "LDPC Encoder and Method

Thereof", filed on December 7, 2000 and assigned application Serial No. 09/730,752, the contents of which are incorporated herein by reference. The parity data from linear block code encoder 304 is combined with the data encoded by encoder 302 by multiplexer 306 for input to channel transmitter 310. Preferrably, the combined data consists of series of a pair parity bits followed by 40 bits of user data. This constraint is established by encoder 302.

[0010] Transmitter 310 transmits the combined user and parity data from multiplexer 306 typically as an analog signal over communication channel 401 in the channel domain. Communication channel 401 may include any wireless, wire, optical, magnetic and the like.

[0011] Receiver 500' comprises an analog to digital converter 502 to convert the data transmitted on communication channel 401 to a digital signal. The digital signal is input to soft channel decoder 504, which provides soft or probabilistic information of the detected data to soft linear block decoder 506. Soft channel decoder may be implemented as a Soft Viterbi Detector or the like, and address generator 530 may be constructed similarly as address generator 328 in transmission section 300'. The soft information output by soft channel decoder 504 remains in the channel domain and is decoded by soft linear block code decoder 506, in accordance with the address of the parity equations generated by address generator 530. Address generator 530 is responsive to counter 735 under the control of controller 745. Controller 745 synchronizes counter 735 to the output of soft channel decoder 504 so that counter 830 can provide a count of the number of bits in a codeword output by soft channel decoder 504 and a count of the number of codewords.

[0012] Soft linear block code decoder 506 operates in combination with soft channel decoder 504 and address generator 530 in an iterative fashion. Soft linear block code decoder is preferably implemented as a low-density parity- check code (LDPC) decoder as described in commonly assigned, copending patent application entitled “LDPC Decoder and Method Thereof”, filed on December 7, 2000 and assigned application Serial No. 09/730603, the contents of which are incorporated herein by reference.

[0013] After the iterative process has completed, the output of soft linear block code decoder 506 is passed on for further processing to decoder 508. Decoder 508 is implemented to perform the reverse operations of encoder 302 or correct for any data errors.

[0014] One type of communication channel for which soft-output decoding of convolutionally encoded data is often employed is the controlled intersymbol interference (ISI) channel. One special class of controlled ISI channels that is commonly used is that of the partial response (PR) channels, which are used to model magnetic recording channels. Because of the redundancy inherent in a convolutional code and the iterative nature of the decoding process, the decoding method used may be highly complex, and therefore consume an unacceptably high level of computing resources. Accordingly, there is a need for a new soft-output decoding method having reduced complexity while continuing to yield low bit error rates for PR channels.

SUMMARY OF THE INVENTION

[0015] The present invention is intended to address the need for a soft-output decoding method having reduced complexity.

[0016] In one aspect, the invention provides a decoder for decoding convolutionally encoded data from a communication channel. The channel is associated with a predetermined plurality of dominant error events. The decoder includes a Viterbi detector, which is configured to detect a survivor path for an input symbol and to make a hard decision about a polarity of the input symbol based on the survivor path. The decoder also includes circuitry configured to determine which of the plurality of dominant error events are possible dominant error events with respect to the input symbol, circuitry configured to determine which of the possible dominant error events has a least penalty metric with respect to the survivor path, and a calculator. The calculator is configured to compute an approximation to a logarithmic likelihood ratio for the input symbol based on the survivor path and the determined possible dominant error event having the least penalty metric. The communication channel may be a controlled intersymbol interference channel, which may be capable of being modeled as a finite impulse response (FIR) filter. The controlled intersymbol interference channel may be a partial response channel, such as either a partial response class-IV (PR4) channel, an extended partial response class-IV (EPR4) channel.

[0017] In another aspect, a read channel for a hard disk drive is provided. The read channel is associated with a predetermined plurality of dominant error events. The read channel includes a decoder for decoding data accessed by the read channel. The decoder includes a Viterbi detector, which is configured to detect a survivor path for an input symbol and to make a hard decision about a polarity of the input symbol based on the survivor path. The decoder also includes circuitry configured to determine which of the plurality of dominant error events are possible dominant error events with respect to the input symbol, circuitry configured to determine which of the possible dominant error events has a least penalty metric with respect to the survivor path, and a calculator configured to compute an approximation to a logarithmic likelihood ratio for the input symbol based on the survivor path and the determined possible dominant error event having the least penalty metric. The read channel may be a controlled intersymbol interference channel, which may be capable of being modeled as a finite impulse response (FIR) filter. The controlled intersymbol interference channel may be a partial response channel, such as either a partial response class-IV (PR4) channel, an extended partial response class-IV (EPR4) channel.

[0018] In yet another aspect, a disk drive for accessing data is provided. The disk drive includes a read channel, which is associated with a predetermined plurality of dominant error events. The read channel includes a decoder for decoding data accessed by the disk drive. The decoder includes a Viterbi detector, which is configured to detect a survivor path for an input symbol and to make a hard decision about a polarity of the input symbol based on the survivor path. The decoder also includes circuitry configured to determine which of the plurality of dominant error events are possible dominant error events with respect to the input symbol, circuitry configured to determine which of the possible dominant error events has a least penalty metric with respect to the survivor path, and a calculator configured to compute an approximation to a logarithmic likelihood ratio for the input symbol based on the survivor path and the determined possible dominant error event having the least penalty metric. The read channel may be a controlled intersymbol interference channel, which may be capable of being modeled as a finite impulse response (FIR) filter. The controlled intersymbol interference channel may be a partial response channel, such as either a partial response class-IV (PR4) channel, an extended partial response class-IV (EPR4) channel.

[0019] In still another aspect, the invention provides an apparatus for reducing complexity in a Viterbi decoding algorithm used in a communication channel. The apparatus includes means for identifying a survivor path for an input symbol, means for making a hard decision about a polarity of the input symbol based on the identified survivor path, means for identifying a plurality of dominant error events for which the opposite polarity would be determined for the input symbol, means for measuring a penalty metric value based on the identified survivor path for each of the plurality of dominant error events, means for choosing a dominant error event having a least penalty metric value from the identified plurality of dominant error events, and means for calculating an approximation to a logarithmic likelihood ratio for the input symbol based on the survivor path and the chosen dominant error event. The communication channel may be a controlled intersymbol interference channel, which may be capable of being modeled as a finite impulse response (FIR) filter. The controlled intersymbol interference channel may be a partial response channel, such as either a partial response class-IV (PR4) channel, an extended partial response class-IV (EPR4) channel.

[0020] In yet another aspect, the invention provides an apparatus for accessing data from a hard disk drive using a read channel. The read channel is associated with a predetermined plurality of dominant error events. The read channel includes a decoder means for decoding data accessed by the apparatus. The decoder means includes a Viterbi detector means for detecting a survivor path for an input symbol and for making a hard decision about a polarity of the input symbol based on the survivor path, means for determining which of the plurality of dominant error events are possible dominant error events with respect to the input symbol, means for determining which of the possible dominant error events has a least penalty metric with respect to the survivor path, and means for calculating an approximation to a logarithmic likelihood ratio for the input symbol based on the survivor path and the determined possible dominant error event having the least penalty metric. The read channel may be a controlled intersymbol interference channel, which may be capable of being modeled as a finite impulse response (FIR) filter. The controlled intersymbol interference channel may be a partial response channel, such as either a partial response class-IV (PR4) channel, an extended partial response class-IV (EPR4) channel.

[0021] In still another aspect of the invention, a method of reducing complexity in a Viterbi decoding algorithm used in a communication channel is provided. The method includes the steps of identifying a survivor path for an input symbol, making a hard decision about a polarity of the input symbol based on the identified survivor path, identifying a plurality of dominant error events for which the opposite polarity would be determined for the input symbol, measuring a penalty metric value based on the identified survivor path for each of the plurality of dominant error events, choosing a dominant error event having a least penalty metric value from the identified plurality of dominant error events, and calculating an approximation to a logarithmic likelihood ratio for the input symbol based on the survivor path and the chosen dominant error event. The communication channel may be a controlled intersymbol interference channel, which may be capable of being modeled as a finite impulse response (FIR) filter. The controlled intersymbol interference channel may be a partial response channel, such as either a partial response class-IV (PR4) channel, an extended partial response class-IV (EPR4) channel.

[0022] In yet another aspect of the invention, a method of accessing data from a hard disk drive using a read channel is provided. The read channel is associated with a predetermined plurality of dominant error events. The method includes the steps of identifying a survivor path for an input symbol, making a hard decision about a polarity of the input symbol based on the identified survivor path, identifying a plurality of dominant error events for which the opposite polarity would be determined for the input symbol, measuring a penalty metric value based on the identified survivor path for each of the plurality of dominant error events, choosing a dominant error event having a least penalty metric value from the identified plurality of dominant error events, and calculating an approximation to a logarithmic likelihood ratio for the input symbol based on the survivor path and the chosen dominant error event. The read channel may be a controlled intersymbol interference channel, which may be capable of being modeled as a finite impulse response (FIR) filter. The controlled intersymbol interference channel may be a partial response channel, such as either a partial response class-IV (PR4) channel, an extended partial response class-IV (EPR4) channel.

[0023] In still another aspect, the invention provides a method of decoding convolutionally encoded data accessed by a read channel for a hard disk drive. The read channel is associated with a predetermined plurality of dominant error events. The method includes the steps of identifying a survivor path for an input symbol, making a hard decision about a polarity of the input symbol based on the identified survivor path, identifying a plurality of dominant error events for which the opposite polarity would be determined for the input symbol, measuring a penalty metric value based on the identified survivor path for each of the plurality of dominant error events, choosing a dominant error event having a least penalty metric value from the identified plurality of dominant error events, and calculating an approximation to a logarithmic likelihood ratio for the input symbol based on the survivor path and the chosen dominant error event. The read channel may be a controlled intersymbol interference channel, which may be capable of being modeled as a finite impulse response (FIR) filter. The controlled intersymbol interference channel may be a partial response channel, such as either a partial response class-IV (PR4) channel, an extended partial response class-IV (EPR4) channel.

[0024] In still another aspect, the invention provides a storage medium for storing software for implementing each of the methods described above. The software is computer-readable.

BRIEF DESCRIPTION OF THE DRAWINGS

[0025] Figure 1 illustrates a survivor path and an error event for a Viterbi detection of a convolutional code.

[0026] Figure 2 shows a block diagram of a finite-impulse response filter model of a controlled ISI channel.

[0027] Figure 3 shows a flow chart of a method for calculating a logarithmic likelihood ratio according to the present invention.

[0028] Figure 4 shows a construction of a set of dominant error events for a particular channel.

[0029] Figure 5 shows a block diagram of a method of calculating the LLR for a given input, according to the present invention.

[0030] Figure 6 is a block diagram of a data transmission system.

DETAILED DESCRIPTION OF THE INVENTION

[0031] Referring to Figure 2, a controlled intersymbol interference (ISI) channel can be modeled as a finite impulse response (FIR) filter, because it has finite response length. Examples of such a channel include the partial response class-IV (PR4) channel (i.e., $1\text{-}D^2$), the extended partial response class-IV (EPR4) channel (i.e., $1\text{+}D\text{-}D^2\text{-}D^3$), the “221” channel ($2\text{+}2D\text{-}D^2\text{-}2D^3\text{-}D^4$), etc., all of which are used to model magnetic recording channels. A controlled ISI channel such as one of these is often used, for example, as a read channel in a disk drive.

[0032] Controlled ISI channels can be represented by trellis. Viterbi detection is commonly used to decode the inputs to such channels. A Viterbi detector tries to find the minimum metric path (referred to as the "survivor path") through the channel trellis. Where the path metric is the squared distance (Euclidean distance) between the noiseless channel output corresponding to the path and the received noisy channel samples. Referring to Figure 1, an error event occurs if the survivor path does not correspond to the true inputs. Such a situation is possible if the true input path does not correspond to the minimum metric path, which can result from noise in the channel output samples.

[0033] Dominant error events are the most likely error events for controlled ISI channels, because the difference metrics between dominant error event paths and the true input path are small. Because of these small difference metrics, it is easy to mistake the dominant error event paths for the true input path. For controlled ISI channels, the dominant error event paths are readily identifiable. For example, for the "221" channel, the dominant error events are "+ - +", "+", and "+ -".

[0034] Turbo codes, LDPC codes, and other convolutional codes require probabilistic (i.e., soft) information for each channel input symbol. Referring to Figure 2, a controlled ISI channel may be modeled as an FIR filter having transfer function H. To obtain the needed soft information, the logarithmic likelihood ratio (LLR) must be calculated using Equation (1) below:

$$(1)\quad LLR(x_k) = \log \frac{P(x_k=+1)}{P(x_k=-1)} = \log \frac{\sum_{\text{all paths such that } x_k=+1} P(\text{each such path})}{\sum_{\text{all paths such that } x_k=-1} P(\text{each such path})}$$

[0035] Traditional soft-decoding methods, such as the BCJR algorithm or the soft-output Viterbi algorithm (SOVA), attempt to find the probabilities for all or most of the paths and then carry out Equation (1). However, for controlled ISI channels, the survivor path and the paths corresponding to the dominant error events are the most likely paths through the trellis. Thus, Equation (1) can be approximated by Equation (2) below for a controlled ISI channel:

$$(2) \quad \log\frac{P(x_k=+1)}{P(x_k=-1)} \approx \log\frac{P(\text{the most likely path such that } x_k=+1)}{P(\text{the most likely path such that } x_k=-1)}$$

Without loss of generality, it may be assumed that the survivor path corresponds to $x_k = +1$. This assumption leads to Equation (3):

$$(3) \quad \log\frac{P(x_k=+1)}{P(x_k=-1)} \approx \log\frac{P(\text{survivor path})}{P(\text{error event with the least penalty metric such that } x_k=-1)}$$

[0036] Therefore, referring to Figure 3, a preferred embodiment of a new soft-output decoding algorithm 350 can be described as follows. The first step 355 is to find the survivor path of the Viterbi detector. The second step 360 is to make a hard decision x_k based on the survivor path. The third step 365 is to then, from all of the dominant error events, identify the ones that would change the polarity (from 1 to 0, or from 0 to 1) of the decision x_k. The fourth step 370 is to then, from among all of the error events so identified, choose the one that has the minimum penalty metric as compared with the survivor path. Finally, the last step 375 is to use Equation (3) to calculate the soft information about input symbol x_k based on the chosen error event.

[0037] In summary, the knowledge about the dominant error events greatly reduces the search space for the paths in Equation (1), therefore making it possible to extract the soft information about x_k without back-tracing along the channel trellis as performed by the BCJR algorithm or the SOVA. As a result, the decoding complexity is greatly reduced, at the expense of slight degradation in performance.

[0038] Referring again to Figure 2, vector x denotes the input to the channel, and the decision vector from the Viterbi survivor path is denoted x_v. The error vector is denoted x_e, therefore, $x_e = x - x_v$, and additive white Gaussian noise (AWGN) is assumed. Therefore, the channel model of Figure 2 yields Equation 4 below:

$$(4) \quad y - (H \otimes (x_v + x_e)) \sim N(0, \sigma^2 I)$$

where $\otimes$ denotes the convolution operation, and I stands for the identity matrix.

[0039] According to the probability density function of the Gaussian noise, the probability of the survivor path x_v is given by Equation (5) below:

$$(5) \quad P(x_v) \propto \exp(-\tfrac{\|y - H \otimes x_v\|^2}{2\sigma^2})$$

Where the squared absolute value term denotes the squared distance of a vector. Equation (5) indicates that large squared distance corresponds to small probability and vice versa. Therefore, squared absolute value term is often referred to as the penalty metric of the path x_v. The Viterbi detector finds the path with the minimum penalty metric, thus the maximum probability.

[0040] Suppose it is desired to calculate the LLR of bit k. Without loss of generality, it is assumed that the Viterbi survivor path x_v corresponds to $x_k = +1$. The set of all error events corresponding to $x_k = -1$ is denoted X_e. For any specific error event $x_e \in X_e$, its probability is given by Equation (6) below:

$$(6) \quad P(x_e) \propto \exp(-\frac{\|y-H\otimes(x_v+x_e)\|^2}{2\sigma^2})$$

If $x_{e,min}$ denotes the error event in X_e that has the largest probability (i.e., the least penalty metric), then Equations (3) and (6) can be combined to produce Equation (7) below:

$$(7) \quad LLR(x_k) = \log\frac{C\cdot\exp(-\frac{\|y-H\otimes x_v\|^2}{2\sigma^2})}{C\cdot\exp(-\frac{\|y-H\otimes(x_v+x_{e,min})\|^2}{2\sigma^2})} = -\frac{1}{2\sigma^2}(\sum_{i=1}^{L}(y_i-(H\otimes x_v)_i)^2 - \sum_{i=1}^{L}(y_i-(H\otimes x_v)_i-(H\otimes x_{e,min})_i)^2)$$

$$= -\frac{1}{2\sigma^2}(\sum_{i,(H\otimes x_{e,min})_i\neq 0}(2(y_i-(H\otimes x_v)_i)\cdot(H\otimes x_{e,min})_i) - \|H\otimes x_{e,min}\|^2)$$

where L is the block length. If the Viterbi survivor path x_v corresponds to $x_k = -1$, then Equation (8) is produced:

$$(8) \quad LLR(x_k) = \frac{1}{2\sigma^2}(\sum_{i,(H\otimes x_{e,min})_i\neq 0}(2(y_i-(H\otimes x_v)_i)\cdot(H\otimes x_{e,min})_i) - \|H\otimes x_{e,min}\|^2)$$

[0041] To implement this algorithm, the set X_e is confined so that it contains only the dominant error events that would change the Viterbi decision at time k. For example, the dominant error events for the H = [2 2 –1 –2 –1] channel are

{(1), (1, -1), (1, -1, 1), (1, -1, 1, -1, 1)}.

Any shift or inverse of the list error events is also a dominant error event, so long as such an error event may possibly occur. Referring to Figure 4, an example is shown using the H = [2 2 –1 –2 –1] channel and its dominant error events to illustrate how to construct the set X_e from the dominant error event list and the Viterbi hard decisions, per step 365 in Figure 3. A sequence of bits 405 represents the Viterbi hard decisions x_v. A particular bit 410 is designated as the k^{th} bit in the sequence 405, and thus it represents x_k.

[0042] To construct the set X_e, as per step 365 in Figure 3, each possible dominant error event, including any shift or inverse of each possible dominant error event, is checked against the x_k bit 410 and its nearest neighboring bits within the x_v sequence 405 to see whether that dominant error event may possibly have occurred. As shown above, the first dominant error event for the H channel is (1), i.e., a one-bit error. As shown in Figure 4, $x_k = 1$, so (1) would not be an error at all. However, the inverse of (1) is (-1), and this would be an error, thus it is a possible error, and it is designated as the first possible dominant error event 415 within the set X_e.

[0043] The second dominant error event for the H channel is (1, –1). This is a two-bit error, so it must be compared with a two-bit sequence within x_v. First, it is checked against (x_k, x_{k+1}): $x_k = 1$ and $x_{k+1} = 0$, so this would not be an error; however, the inverse of this dominant error event is (–1, 1), and this would be a possible error, Thus, it is designated as the second possible dominant error event 420 within the set X_e. Then, the same event is checked against the sequence of (x_{k-1}, x_k), and it is found that (1, –1) is a possible dominant error corresponding to that sequence, and it is designated as the third possible dominant error event 425 within the set X_e. The next dominant error event for the H channel is (1, –1, 1). This is a three-bit error, so it can be shifted into three possible positions that include x_k, and it can also be inverted, so there are six possibilities to be checked. After checking, it is seen that the (–1, 1, –1) event 430 is a possibility for the sequence of (x_k, x_{k+1}, x_{k+2}), and the (1, –1, 1) event 435 is a possibility for the

(x_{k-1}, x_k, x_{k+1}) sequence. Finally, the last event to be checked is the (1, –1, 1, –1, 1) five-bit event. This event has ten permutations, including five shift positions having two inverses each. After checking, it is seen that it is impossible for any permutation of this event to occur in such a way that x_k is included. Thus, the set X_e is complete, and its elements include the five dominant error events 415, 420, 425, 430, 435 that are possibilities corresponding to x_k.

[0044] Referring to Figure 5, a block diagram 500 shows how to calculate the LLR value based on X_e, the channel output y, and the Viterbi decision x_v, as per steps 370 and 375 in Figure 3. Continuing with the example described above with reference also to Figure 4, each of the five elements of the set X_e are plugged into Equation (6), as indicated by the blocks 505 showing the calculation of LLR(x_k) for each respective element of X_e. Once these five calculations are completed, the minimum of these five is found, per block 510. This corresponds to the dominant error event having the least penalty metric. Finally, this result is multiplied by the sign of the survivor path itself, as per Equation (8), to produce LLR(x_k), which is the desired soft information corresponding to x_k.

[0045] The method described herein reduces the complexity of the Viterbi decoding algorithm by approximating the summations in the numerator and the denominator of Equation (1) using only the survivor path and the dominant error event having the least penalty metric, respectively. The tradeoff for these approximations is a slight degradation in performance. It is noted that a preferred embodiment of the invention is an application of the decoding method described above in a read channel in a hard disk drive. Experimentation with such a read channel implementation has shown that this performance degradation is equivalent to a reduction in the signal-to-noise ratio (SNR) of about 0.3 dB. However, because the calculation is so much simpler, there is a tremendous savings in memory, in that the state memory is not needed at all. Furthermore, there is also an energy savings of about 50% as compared to the energy consumption for performing the full calculation of Equation (1).

[0046] While the present invention has been described with respect to what is presently considered to be the preferred embodiment, it is to be understood that the invention is not limited to the disclosed embodiments. To the contrary, the invention is intended to cover various modifications and equivalent arrangements included within the spirit and scope of the appended claims. For example, it is to be understood that the invention is applicable to communications channels other than controlled ISI channels that use soft-output decoding schemes. The scope of the following claims is to be accorded the broadest interpretation so as to encompass all such modifications and equivalent structures and functions.